Journal Intelligence Creation and Application Engineering

Website: https://ioinformatic.org/

date 15 October 2025. Vol. 5 . No. 1; e-ISSN: 2808-4519

Implementation of Base 64 and AES Algorithms in Web-Based Email Message Security System

Indri Renika[1] *, Rahmadani [2], I Gusti Prahmana [3]

[1,2,3] STMIK Kaputama

indrirenika@gmail.com[1], rahm4dani@gmail.com[2] , igustiprahmana4@gmail.com[3]*

Abstract

This research is motivated by the increasing threat to message security in email communications due to the rapid development of information technology. To address this, this study aims to implement and test a web-based email message security system that combines Base64 and AES 256-bit algorithms to protect text messages and attachments from security threats. This system was developed using the PHP programming language and functions as an internal messaging system. The results show that the combination of the two algorithms successfully creates a system capable of securing messages strongly. During the delivery process, the system performs AES 256 encryption and Base64 *encoding* on messages and attachments. Meanwhile, when a message is received, the user must first enter the same key, after which the system will perform Base64 *decoding* and continue with AES 256 decryption to restore the message to its original form. Thus, the resulting system is proven effective in securing digital communications and ensuring that messages can only be accessed by authorized recipients.

Keywords: *AES 256 bit algorith , Base64 algorithm , Email messages , Security Message*

1. Introduction

Rapidly development technology information make email as tool exchange information distance far away is important for individual and business . However , the popularity of email also triggered improvement significant to the threat data security , especially targeting security message text and messages images that are often attached or sent via email. Attack like wiretapping that can reveal content message text sensitive , theft identity through information in images , and data manipulation that changes content message become threat Serious to privacy and integrity information . One of the method For increase security is use technique cryptography .

Study This propose merging two pieces method cryptography namely Base64 and AES *(Advanced Encryption Standard),* as effort in secure message text and messages image sent via email [1]. Research conducted by has apply AES 128 bit and Base64 algorithms . Although AES 128 bit provides level significant security , long the key is more short make it potential more prone to to brute-force attack , namely method break in security with try all possibility key or password one by one until successful , with progress computing moment this and the threats in the future , especially compared to with standard more security recommended height long more keys big . In the study This writer will make AES algorithm with 256 bits and Base64. Research This important Because with adding bits to the AES algorithm then will increase security encryption on email messages significant compared to with AES128-bit, so that make message more difficult solved by someone else authorized. In addition to the AES research algorithm this will also.use algorithm Base64 cryptography as encoding and decoding.

In research This will encoding and decoding is done using Base64 algorithm . With implementing Base64, the data sent can avoided from problem invalid characters and ensure that the data remains intact during the delivery process . Base64 according to [2] is one of the algorithm for encoding and decoding binary data in ASCII format , which is basically is the number 64 or can it is said as one of the method encoding on binary data. Research this will also uses AES *(Advanced Encryption Standard)* which complies [3] A method cryptography that uses algorithm cipher block . This method also involves technique substitution , permutation , and some rounds on each the block that will encrypted and decrypted .

Referring to the importance of security in communication via email, research This develop AES algorithm with 256 bit key , a improvement significant from study previously only using AES 128 bit.with long more keys large , AES 256 bit offers level distant security more tall to attack like wiretapping , theft identity , and data manipulation in email messages . With blend Base64 and AES algorithms , it is expected created A a system that does not only protect data from threat external , but also ensure that the data sent still whole and only can accessed by authorized recipients .
Study This produce system strong email security , capable of protect message text and images from various threat . System This using two techniques algorithm that is , the AES algorithm will encrypt message then Base64 will encoding results AES encryption into text format

, the Base64 decoding process will done on the side recipient before AES decryption returns the data to its original format . After it is the decryption process that will done and recipient will accept message original with safe use the key that has been determined .

2. Methodology Study

Research methods is a series step systematically used For collecting , analyzing , and interpreting data. Research methods used in various discipline knowledge and can varies depending on the purpose research , nature and data. The following is a methodology diagram research used in the study moment This :

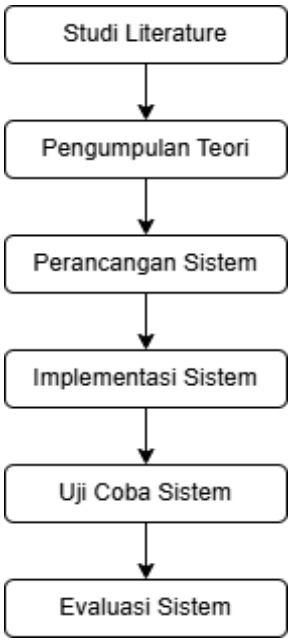


Fig. 1: Methodology Study

From the flow diagram in the image above, the stages of this research can be described as follows: following:

1. Literature Study : Stages This is a search process deep to various source scientific like relevant journals , books , and articles . The focus is understand concepts base related security messaging , cryptography , and development system web -based which becomes foundation this research .
2. Collection : After studies literature , concepts and core theories that become runway main system collected and summarized . Theories This covers principle Work algorithm AES 256-bit and Base64 cryptography will implemented .
3. Design System : At stage this , structure and flow Work system planned . Design This covering architecture system , interface user interface (UI/UX), and the database schema that becomes guide in stage development system .
4. Implementation System : This is the development process system security email message . Plans that have been made made realized in form program code using Language PHP programming for build all over functionality system .
5. System Testing : After system finished built , a series of testing done For ensure that every features , especially the encryption and decryption processes , function with right . Stage this also ensures that system can walk in accordance with expected functionality .
6. Evaluation System : At stage lastly , the results from all over testing analyzed . From the analysis this , can concluded whether system security email message successful reach the goal For protect messages , and identify potential future improvements .

3. Results and Discussion

3.1. Discussion

System security web -based email messaging with integrate AES 256-bit and Base64 algorithms. Combination second algorithm This proven effective ; AES 256-bit provides level strong security , while Base64 ensures encrypted data integrity For safe delivery and storage. All trials functionality , starting from management account users until the delivery process and decryption message , shows positive results. With Thus , the system developed This succeed reach objective study in provide solution practical and reliable For protect digital message from threat security.

3.2. Implementation

Study This aim For implement system security web -based email messages that combine AES 256-bit and Base64 algorithms . Based on structured methodology , system succeed implemented using PHP with functionality complete , started from management account users like registration and login up to feature delivery message encrypted . The entire encryption and decryption process verified in a way technical with display stages per round of the AES-256 (ECB) algorithm , including validation 32-byte key and usage PKCS#7 *padding* . Encryption results then encoded with Base64 for ensure data security moment saved , while the decryption process is going on proven

capable return messages and attachments to form the original after key entered . All test results , including recording messages in history out , prove that system This effective and reliable in secure digital communication .

3.3. System Trial

After phase implementation , carried out a series testing For verify that all over **feature** system Work in accordance with what is expected Testing This done in a way in stages , including :

1. Trial Account Registration

The account registration test is a testing phase to verify the functionality of registering a new account on the system. As shown in the image, the test is performed by entering *the username* user1@gmail.com and password. After clicking the "Register" button, the system successfully processes the data and displays a confirmation message: "Account created successfully. Please log in." This confirms that the account registration process is functioning properly.

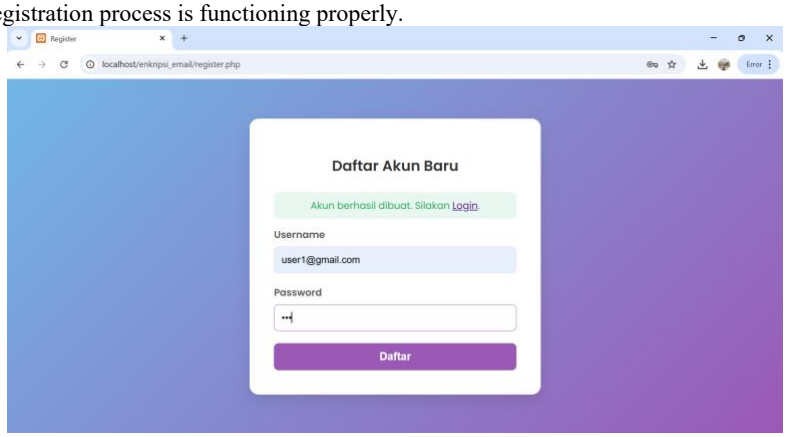


Fig. 2: Trial account list

2. User Account Login Trial

User account login testing is performed to verify that users can successfully log into the system using valid login information. The image shows *the username* user1@gmail.com and the previously registered password entered into the login form. This successful process demonstrates that the login system is functioning properly and is ready for use by users.

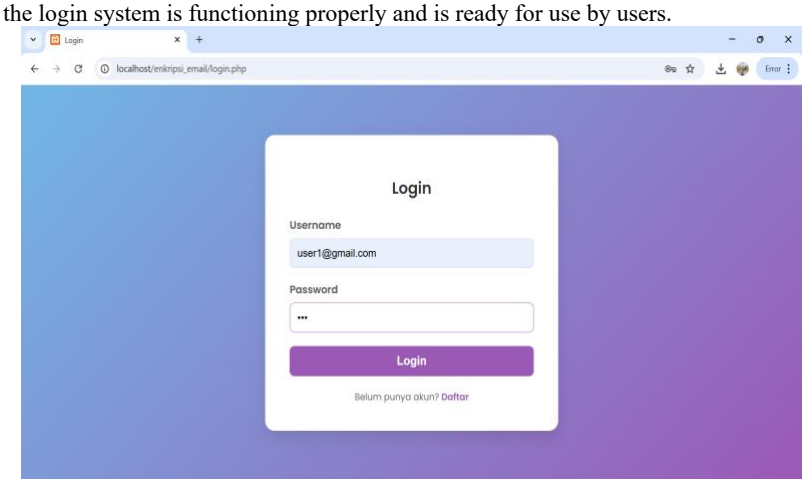


Fig. 3: login test

3. Trial Sending Message Encrypted

Encrypted messaging testing was conducted to verify the system's core functionality in securing messages. In this test, the user filled out a form with the recipient's details (user2@gmail.com), subject, and message content, and attached the PERSETU...OPOSAL.pdf file. Before the message was sent, the user entered an encryption key, which is essential for the security process. The success of this step confirmed that the system was able to encrypt and securely send data, including attachments, as designed.

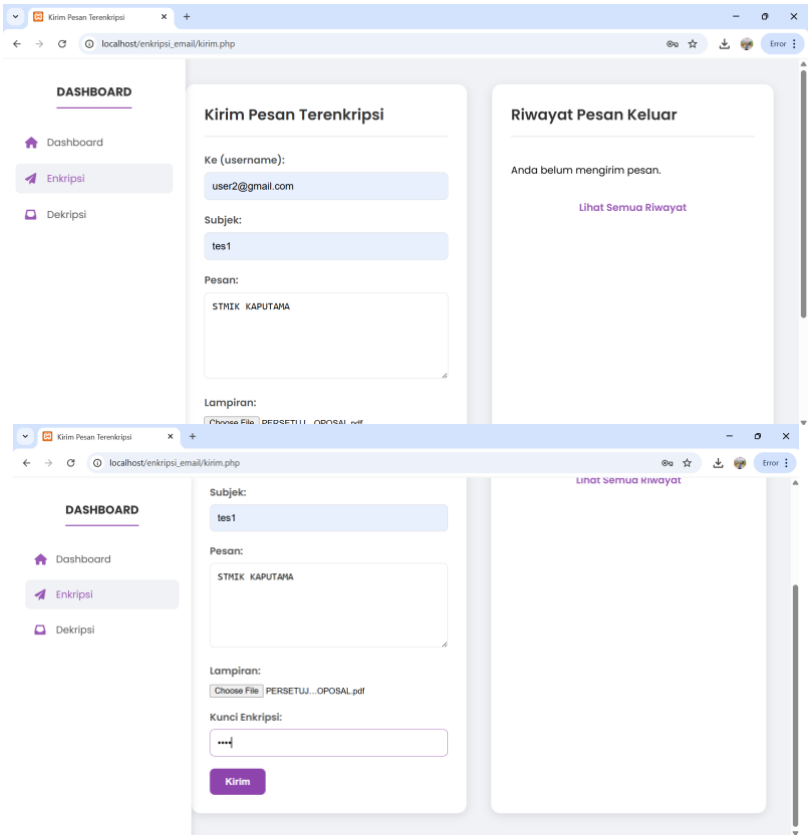


Fig. 4: trial message encrypted

4. Display Test Select Attachment File

The attachment file selection display test was conducted to verify that the user interface functions properly when selecting files to attach. The image shows that the system successfully displays a *file explorer window* , allowing users to navigate files on their computer. For example, the user successfully selects the image file 'nature-green-forest-589967.jpg' to upload, indicating that this feature has been successfully implemented.

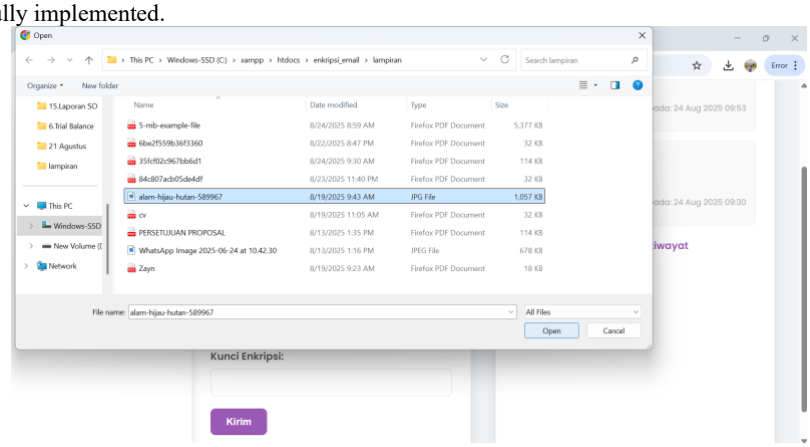
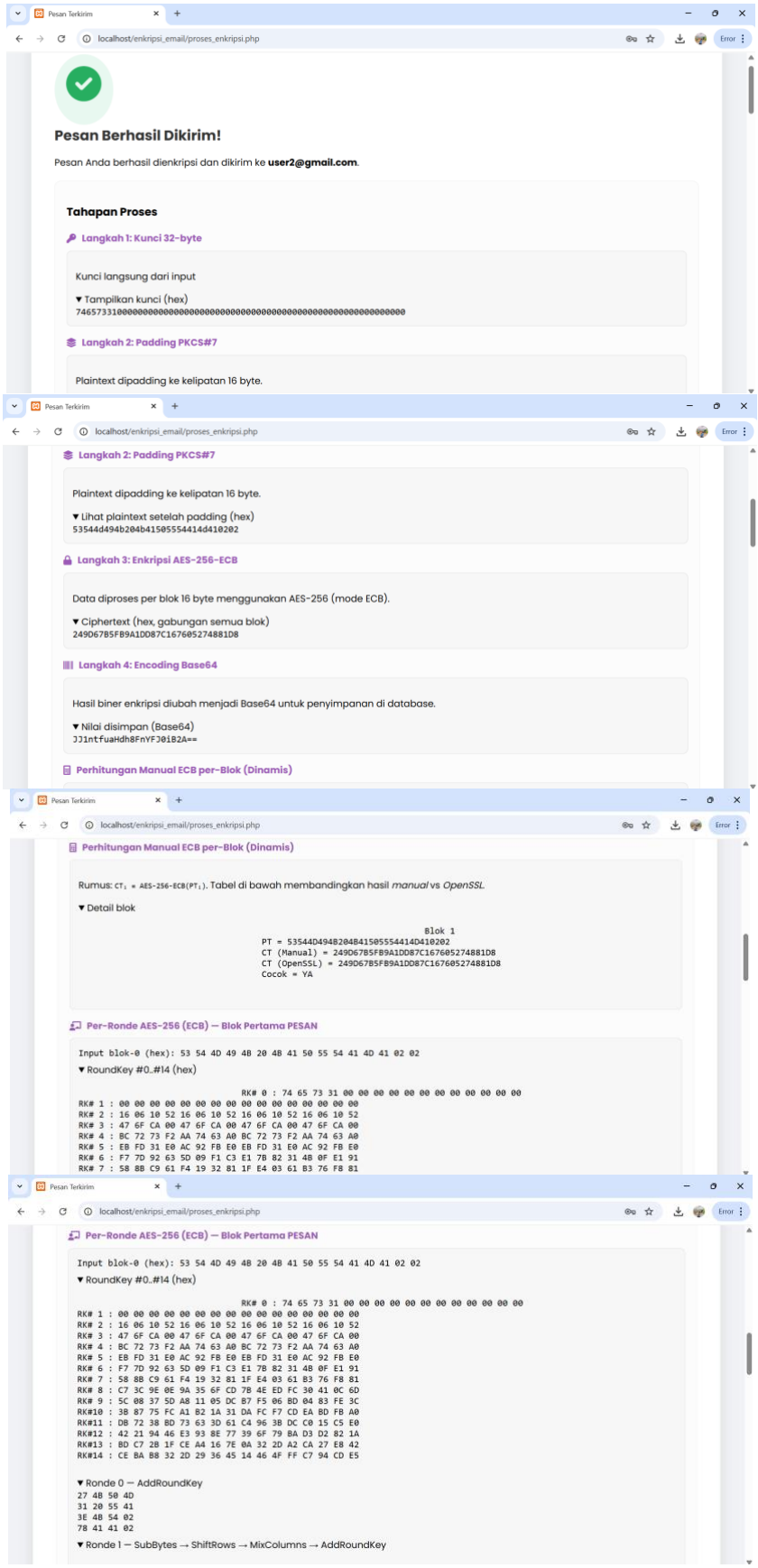
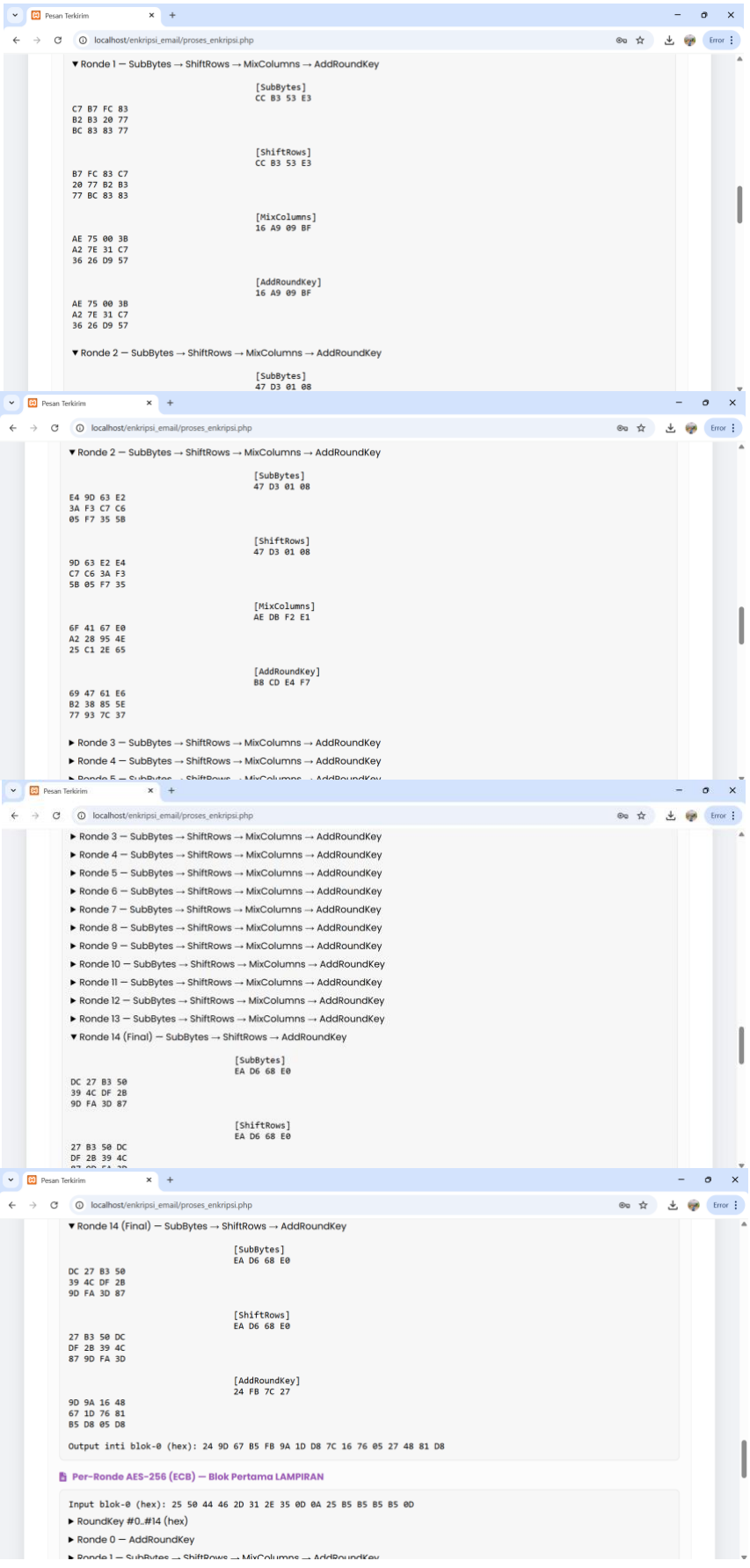


Fig. 5: trial select the attached file

5. Test Encryption Process Result Display

This view shows the technical details of the successful encrypted message delivery process. After the user sends a message and attaches a file, the system initiates a series of encryption steps . picture following :


Pesan Terkirim
localhost/enkripsi_email/proses_enkripsi.php
Error
Pesan Berhasil Dikirim!
Pesan Anda berhasil dienkripsi dan dikirim ke user2@gmail.com.
Tahapan Proses
Langkah 1: Kunci 32-byte
Kunci langsung dari input
▼ Tampilkan kunci (hex)
7465733100000000000000000000000000000000000000000000000000000000
Langkah 2: Padding PKCS#7
Plaintext dipadding ke kelipatan 16 byte.
Langkah 2: Padding PKCS#7
Plaintext dipadding ke kelipatan 16 byte.
▼ Lihat plaintext setelah padding (hex)
53544d494b204b41505554414d410202
Langkah 3: Enkripsi AES-256-ECB
Data diproses per blok 16 byte menggunakan AES-256 (mode ECB).
▼ Ciphertext (hex, gabungan semua blok)
249D67B5FB9A1DD87C16760527488 1D8
Langkah 4: Encoding Base64
Hasil biner enkripsi diubah menjadi Base64 untuk penyimpanan di database.
▼ Nilai disimpan (Base64)
JJ1ntfuaHdh8FnYFJ0iB2A==
Perhitungan Manual ECB per-Blok (Dinamis)
Perhitungan Manual ECB per-Blok (Dinamis)
Rumus: CT₁ = AES-256-ECB(PT₁). Tabel di bawah membandingkan hasil manual vs OpenSSL.
▼ Detail blok
Blok 1
PT = 53544D494B204B41505554414D410202
CT (Manual) = 249D67B5FB9A1DD87C16760527488 1D8
CT (OpenSSL) = 249D67B5FB9A1DD87C16760527488 1D8
Cocok = YA
Per-Ronde AES-256 (ECB) — Blok Pertama PESAN
Input blok-0 (hex): 53 54 4D 49 4B 20 4B 41 50 55 54 41 4D 41 02 02
▼ RoundKey #0..#14 (hex)
RK# 0 : 74 65 73 31 00 00 00 00 00 00 00 00 00 00 00 00
RK# 1 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
RK# 2 : 16 06 10 52 16 06 10 52 16 06 10 52 16 06 10 52
RK# 3 : 47 6F CA 00 47 6F CA 00 47 6F CA 00 47 6F CA 00
RK# 4 : BC 72 73 F2 AA 74 63 A0 BC 72 73 F2 AA 74 63 A0
RK# 5 : EB FD 31 E0 AC 92 FB E0 EB FD 31 E0 AC 92 FB E0
RK# 6 : F7 7D 92 63 5D 09 F1 C3 E1 7B 82 31 4B 0F E1 91
RK# 7 : 58 8B C9 61 F4 19 32 81 1F E4 03 61 B3 76 F8 81
Per-Ronde AES-256 (ECB) — Blok Pertama PESAN
Input blok-0 (hex): 53 54 4D 49 4B 20 4B 41 50 55 54 41 4D 41 02 02
▼ RoundKey #0..#14 (hex)
RK# 0 : 74 65 73 31 00 00 00 00 00 00 00 00 00 00 00 00
RK# 1 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
RK# 2 : 16 06 10 52 16 06 10 52 16 06 10 52 16 06 10 52
RK# 3 : 47 6F CA 00 47 6F CA 00 47 6F CA 00 47 6F CA 00
RK# 4 : BC 72 73 F2 AA 74 63 A0 BC 72 73 F2 AA 74 63 A0
RK# 5 : EB FD 31 E0 AC 92 FB E0 EB FD 31 E0 AC 92 FB E0
RK# 6 : F7 7D 92 63 5D 09 F1 C3 E1 7B 82 31 4B 0F E1 91
RK# 7 : 58 8B C9 61 F4 19 32 81 1F E4 03 61 B3 76 F8 81
RK# 8 : C7 3C 9E 0E 9A 35 6F CD 7B 4E ED FC 30 41 0C 6D
RK# 9 : 5C 08 37 5D A8 11 05 DC B7 F5 06 BD 04 83 FE 3C
RK#10 : 3B 87 75 FC A1 B2 1A 31 DA FC F7 CD EA BD FB A0
RK#11 : DB 72 38 BD 73 63 3D 61 C4 96 3B DC C0 15 C5 E0
RK#12 : 42 21 94 46 E3 93 8E 77 39 6F 79 BA D3 D2 82 1A
RK#13 : BD C7 2B 1F CE A4 16 7E 0A 32 2D A2 CA 27 E8 42
RK#14 : CE BA B8 32 2D 29 36 45 14 46 4F FF C7 94 CD E5
▼ Ronde 0 — AddRoundKey
27 4B 50 4D
31 20 55 41
3E 4B 54 02
78 41 41 02
▼ Ronde 1 — SubBytes → ShiftRows → MixColumns → AddRoundKey


Pesan Terkirim
localhost/enkripsi_email/proses_enkripsi.php
Error
▼ Ronde 1 — SubBytes → ShiftRows → MixColumns → AddRoundKey
[SubBytes]
CC B3 53 E3
C7 B7 FC 83
B2 B3 20 77
BC 83 83 77
[ShiftRows]
CC B3 53 E3
B7 FC 83 C7
20 77 B2 B3
77 BC 83 83
[MixColumns]
16 A9 09 BF
AE 75 00 3B
A2 7E 31 C7
36 26 D9 57
[AddRoundKey]
16 A9 09 BF
AE 75 00 3B
A2 7E 31 C7
36 26 D9 57
▼ Ronde 2 — SubBytes → ShiftRows → MixColumns → AddRoundKey
[SubBytes]
47 D3 01 08
E4 9D 63 E2
3A F3 C7 C6
05 F7 35 5B
[ShiftRows]
47 D3 01 08
9D 63 E2 E4
C7 C6 3A F3
5B 05 F7 35
[MixColumns]
AE DB F2 E1
6F 41 67 E0
A2 28 95 4E
25 C1 2E 65
[AddRoundKey]
B8 CD E4 F7
69 47 61 E6
B2 38 85 5E
77 93 7C 37
▶ Ronde 3 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 4 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 5 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 6 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 7 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 8 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 9 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 10 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 11 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 12 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▶ Ronde 13 — SubBytes → ShiftRows → MixColumns → AddRoundKey
▼ Ronde 14 (Final) — SubBytes → ShiftRows → AddRoundKey
[SubBytes]
EA D6 68 E0
DC 27 B3 50
39 4C DF 2B
9D FA 3D 87
[ShiftRows]
EA D6 68 E0
27 B3 50 DC
DF 2B 39 4C
87 9D FA 3D
[AddRoundKey]
24 FB 7C 27
9D 9A 16 48
67 1D 76 81
B5 D8 05 D8
Output inti blok-0 (hex): 24 9D 67 B5 FB 9A 1D D8 7C 16 76 05 27 48 81 D8
Per-Ronde AES-256 (ECB) — Blok Pertama LAMPIRAN
Input blok-0 (hex): 25 50 44 46 2D 31 2E 35 0D 0A 25 B5 B5 B5 B5 0D
▶ RoundKey #0..#14 (hex)
▶ Ronde 0 — AddRoundKey

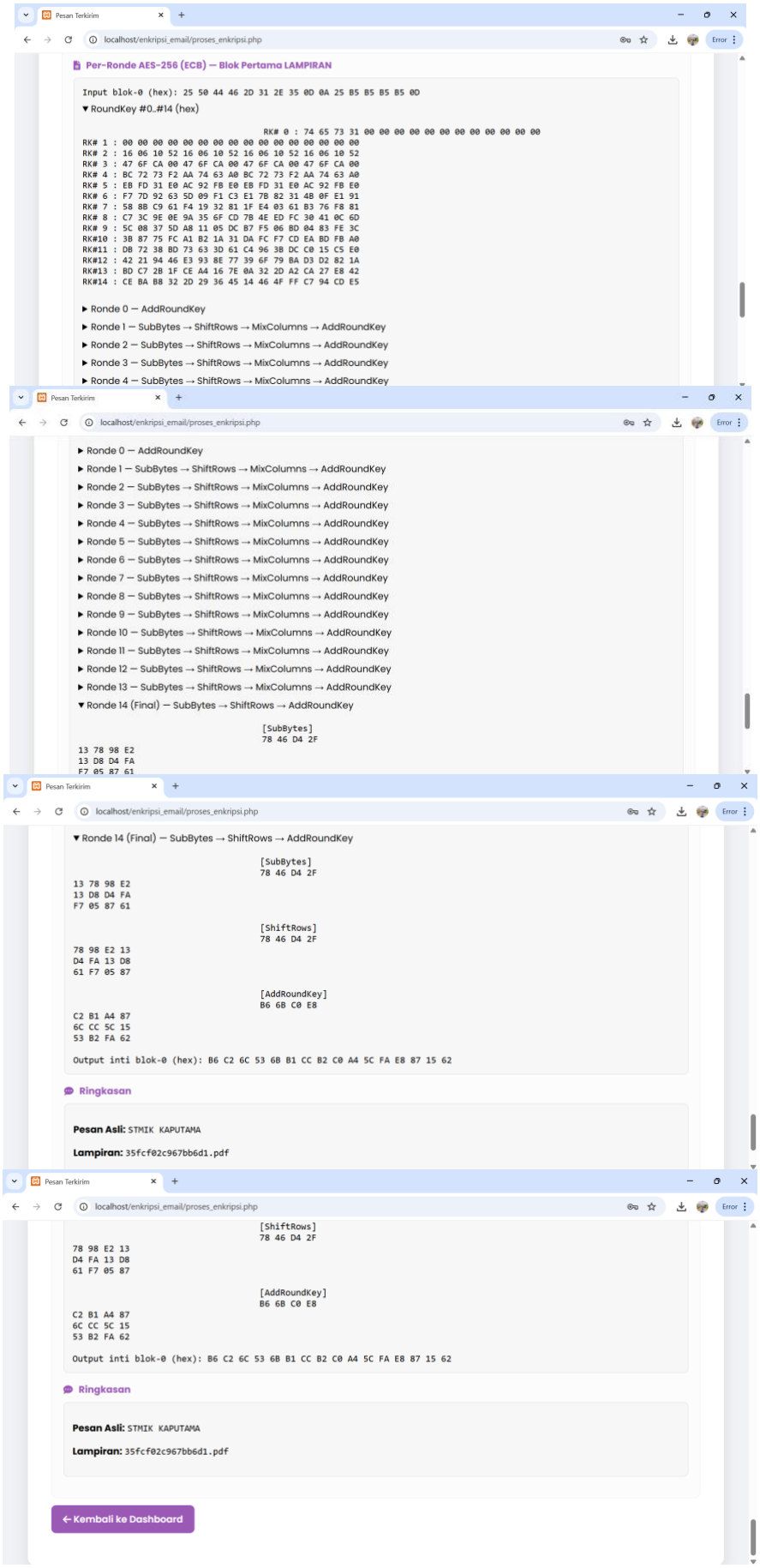
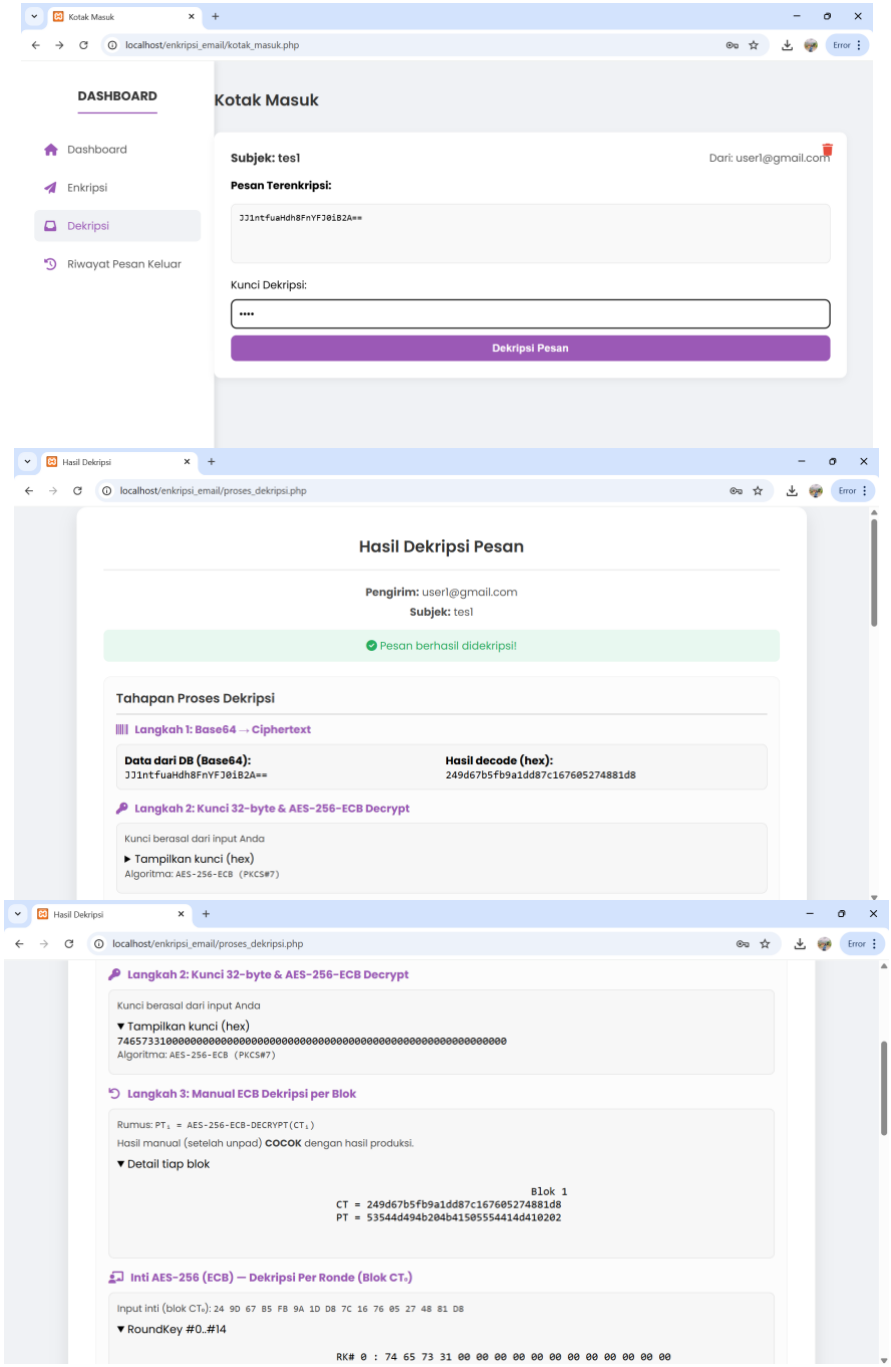


Fig. 6: trial encryption process results

6. Test Display of Decryption Process Results

Testing the decryption results is a crucial step in proving the system's ability to restore data to its original state after encryption. The process begins in the Inbox, where the user selects an encrypted message and enters the decryption key. The system then reverses all encryption steps, from Base64 *decoding* to reverse processing the AES-256 algorithm. The final result is a display containing the recovered original message and attached files, demonstrating the system's excellent ability to maintain data confidentiality and integrity.



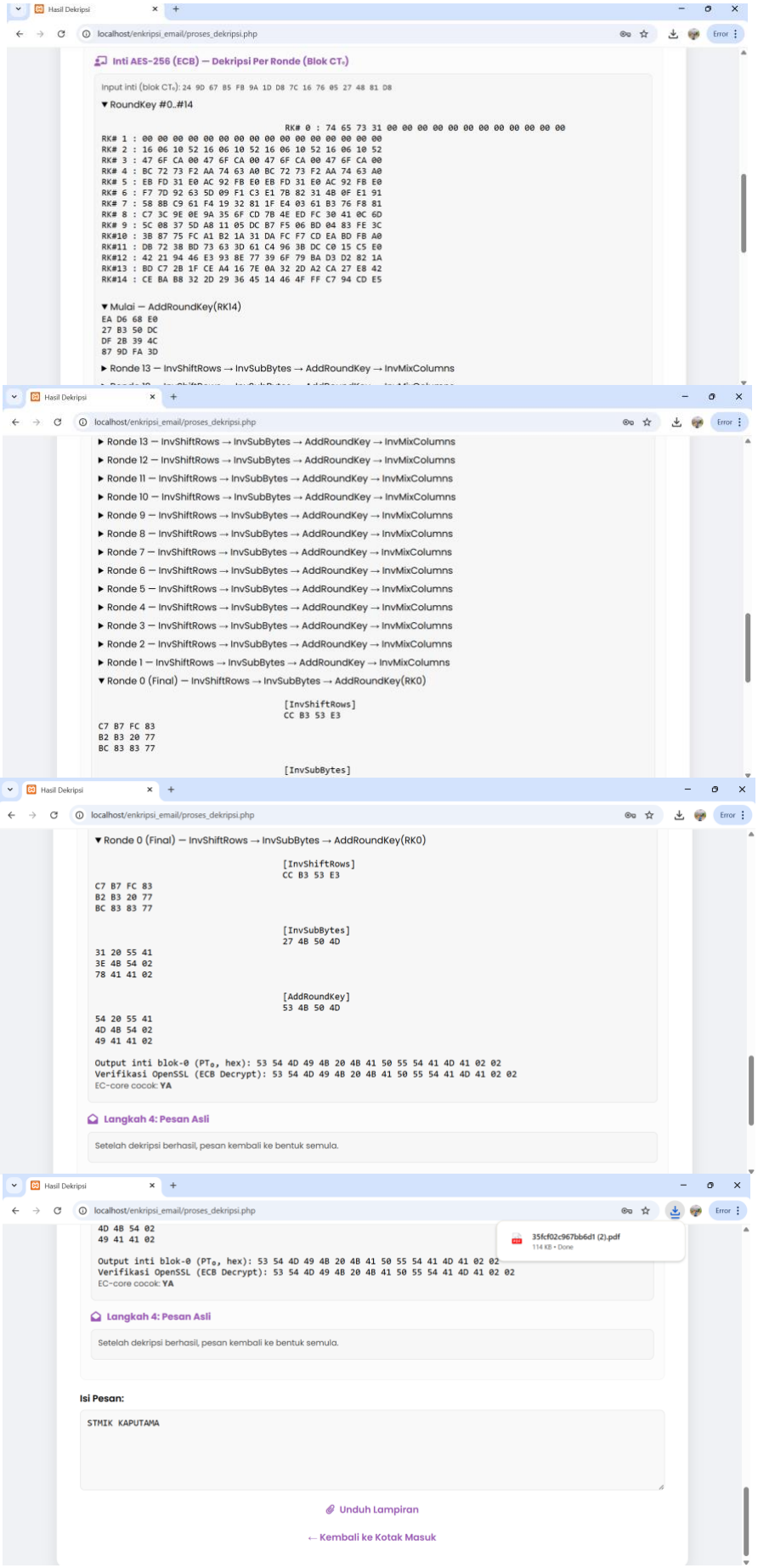


Fig. 7 : trial decryption process results

7. Message Page Test Go out

The outgoing message page aims to verify that the system has successfully saved and displayed the history of messages that have been sent by the user.

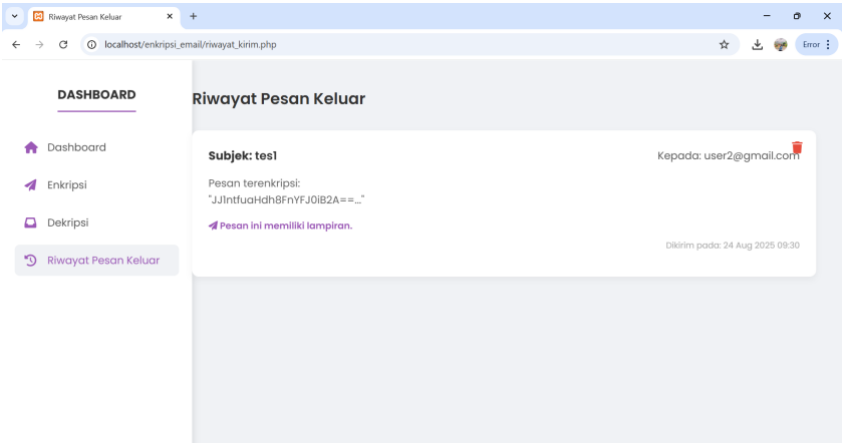


Fig. 8: trial message go out

8. Message Database Input Test

Following This is database view of stored message data through PhpMyadmin like picture following :

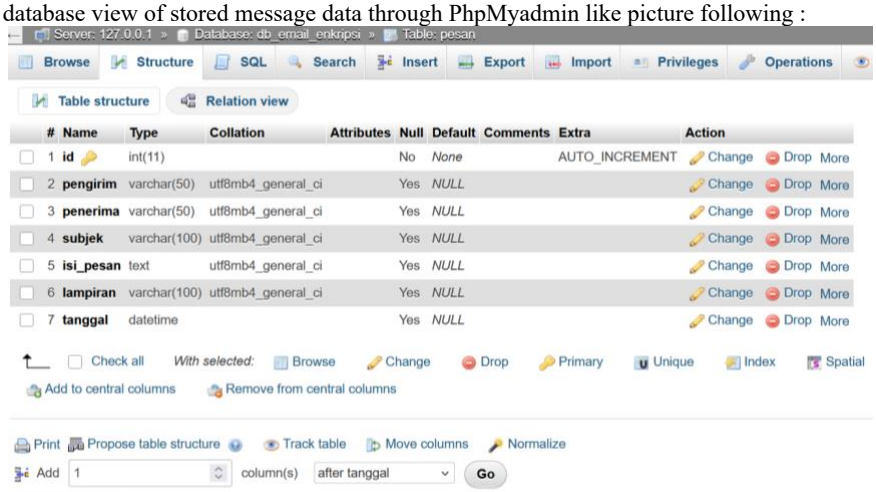


Fig. 9: Message database trial

9. Database Users Input Trial

Following This is database view of stored user data through PhpMyadmin like picture following :

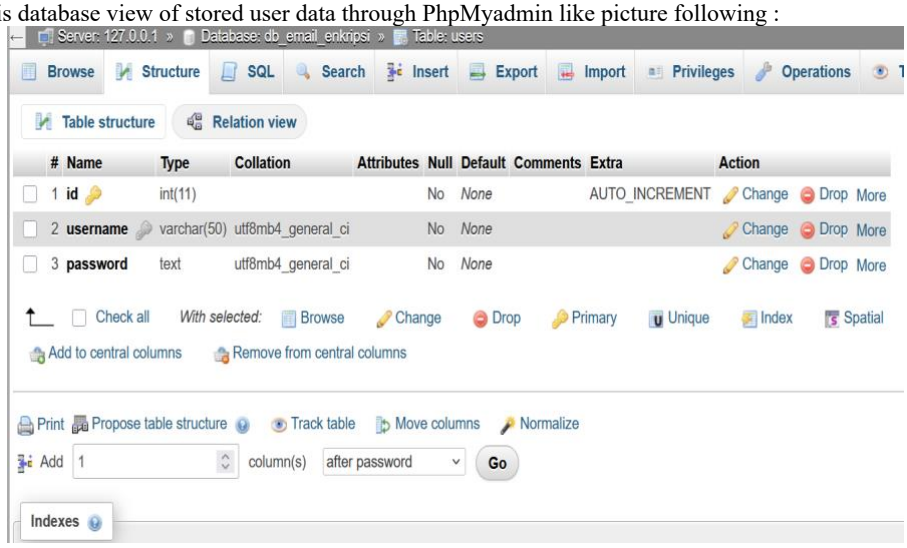


Fig. 10: User database trial

4. Conclusion

Based on all stages of design, implementation, and system testing that have been carried out, it can be concluded that this research has succeeded in building a well-functioning email message security system. Through implementation using the PHP programming language and the phpMyAdmin database, this system is able to apply the AES 256 and Base64 encryption algorithms effectively to secure messages. The test results show that all core functionality runs as expected, starting from the account registration and *login process* , sending messages that are successfully encrypted along with their attached files, to the accurate decryption process to return the message to its original form. Thus, the developed system is proven to be able to provide a strong and reliable solution to maintain the confidentiality and integrity of user messages .

Confession

To all over the party that has give support , guidance , and prayers , I say accept with the greatest love . Hopefully all kindness replied with more Good .

References

[1.] SA Dimas Bayu Nurcahyo., "Dimas Bayu Nurcahyo., & Safrina Amini (2018), Implementation of Cryptography with Base64 Algorithm and Advance Encryption Standard to Secure Web-Based Email Data," 2018.

[2.] GWRF &. AV (. IAA 2. CB 6. DS 2. DPDVDUO Ferzha Putra Utama., "Ferzha Putra Utama., Gusman Wijaya., Ruvita Faurina., & Arie Vatresia4 (2023), Implementation of Aes 256 Cbc, Base 64, and Sha 256 Algorithms in Securing and Validating Online Exam Data," 2023.

[3.] P. Agung Docman Priatama1., "Agung Docman Priatama1., & Painem (2023), Securing Polri Member Data Files Using Aes-128 and Base64 at the Bogor Police Research and Development Center Security of Polri Member Data Files Using Aes-128 and Base64 Method, Bogor Police Research and Development Center," 2023.

[4.] A. Karmakar, S. S. Roy, F. Vercauteren, and I. Verbauwhede, "Efficient finite field multiplication for isogeny based post quantum cryptography," 2017, doi: 10.1007/978-3-319-55227-9_14.

[5.] A. M. H. Pardede, M. Zarlis, and H. Mawengkang, "Optimization of Health Care Services with Limited Resources," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 4, pp. 1444–1449, 2019, doi: 10.18517/ijaseit.9.4.8348.

[6.] A. M. H. Pardede, Y. Maulita, and R. Buaton, "Application modeling ipv6 (internet protocol version 6) on e-id card for identification number for effectiveness and efficiency of registration process identification of population," in *Journal of Physics: Conference Series*, 2018, vol. 978, no. 1, doi: 10.1088/1742-6596/978/1/012017.

[7.] S. P. Mohanty, U. Choppali, and E. Kougianos, "Everything you wanted to know about smart cities," *IEEE Consum. Electron. Mag.*, vol. 5, no. 3, pp. 60–70, 2016, doi: 10.1109/MCE.2016.2556879.

[8.] W. A. Jabbar, W. K. Saad, and M. Ismail, "MEQSA-OLSRv2: A multicriteria-based hybrid multipath protocol for energy-efficient and QoS-aware data routing in MANET-WSN convergence scenarios of IoT," *IEEE Access*, 2018, doi: 10.1109/ACCESS.2018.2882853.

[9.] D. Niyigena, C. Habineza, and T. S. Ustun, "Computer-based smart energy management system for rural health centers," 2016, doi: 10.1109/IRSEC.2015.7455005.

[10.] F.-Z. Younsi, A. Bounnekar, D. Hamdadou, and O. Boussaid, "SEIR-SW, Simulation Model of Influenza Spread Based on the Small World Network," *Tsinghua Sci. Technol.*, vol. 20, no. 5, pp. 460–473, 2015.